A CONTRIBUTION TO THE OPTIMAL NUMBERING OF TREE STRUCTURES

A. F. Martin and E. Alarcón

Cátedra de Estructuras, Escuela Ingenieros Industriales, Madrid, Spain

Abstract—This paper develops an automatic procedure for the optimal numbering of members and nodes in tree structures. With it the stiffness matrix is optimally conditioned either if a direct solution algorithm or a frontal one is used to solvê the system of equations. In spite of its effectiveness, the procedure is strikingly simple and so is the computer program shown below.

1. INTRODUCTION

There are a great number of methods for the automatic mesh generation in F.E.M. and several algorithms for the numbering of nodes and elements. These algorithms are, if not optimum (the concept of optimality is strongly problem dependent), at least very effective in respect of the core space used to store the stiffness matrix and the computations done on it.

Nevertheless, nearly all the procedures are very general and prepared to act on tridimensional structures (either of monodimensional members or of continuum elements) with arbitrary geometry. This is why they are ineffective when treating structures with special features. This is especially true for tree structures mainly used in pipe networks; i.e. the process pipes of chemical industries or the control ones in every highly automatized industry and especially in Nuclear Power Plants.

These kinds of structures present two particular characteristics:

(1) A great number of different elements and components.

(2) The simplicity of its path, fundamentally composed of a series of branches, nearly independent of one another.

Feature (1) is undesirable at description level, due to the vast imput needed to specify all the structural particularities. It would be useful to look for a simple description of the structural properties in order to spend as little time as possible on data preparation and to avoid description errors. This can be accomplished while maintaining the users' freedom of identification (numbering) and grouping of different data.

For instance, an input of the kind used in computer codes prepared for the analysis of general shape structures (such as the STRESS or STRUDL, useful in building structures) is very time-consuming in pipe networks; with its use, it would be necessary, e.g. to define the geometry (node coordinates) separately from the element properties (connectivity and member characteristics).

In this case, it is much more convenient to use a code designed especially for pipe analysis (NUPIPE, for instance) in which the different components, geometry and boundary conditions are defined in a mixed fashion, grouping them by localized pieces of the structure (tree branches). As a natural result of this imput freedom, the initial numbering of members and nodes is randomly organized with respect to a full exploitation of computer storage and capabilities, and this is why an optimization of the numbering is necessary.

Feature (2) is a good property with respect to a good initial numbering, and this is the advantage lost when one uses a generally oriented code. As we shall see below, if we use this second property, we can enormously simplify the algorithms and get, with great simplicity, a very effective tool.

2. DESCRIPTION OF THE PROPOSED METHOD

We start with a special way of presenting the structure, that is, with all data grouped by branches, and with arbitrary node numbering. On the other hand, the members have no number but are identified with the input sequence inside every branch. So we have a set of data in which the connectivities (NCONE – see Glossary at the end of the paper) of every bar are defined. There is also a table of the different trees in which the following properties are defined: (1) number of the node at which every branch starts (I TABLE (I.1)); (2) number of the final branch node (I TABLE (I.3)) and (3) number of the first element in each branch (I TABLE (I.2)).

The results will be, precisely, the new numbering of nodes and members of the structure. The storage of both properties is done in two sets. The first one (IDEN) stores the new node numbering, which can be done either in total form (IDEN (NNUDOS)) or particularized for every degree of freedom of the node (IDEN (NNUDOS, 6)). This last property allows one to eliminate the support-restricted degrees of freedom, with the corresponding space savings.

The second set of results (LM) stores the member of renumbering and can be enlarged with the final numbering of the degrees of freedom of the initial and final nodes of every bar (LM (NBAR, 13)). This facilitates the subsequent assemblage of the total stiffness matrix.

The renumbering procedure consists of taking the first branch and sequentially numbering all its members and nodes until a bifurcation is detected. From here on, and in order to maintain optimal numbering, one must simultaneously number the member and nodes of the first branch and all the others starting at the detected bifurcation. Numbering progresses in a similar fashion, always checking the bifurcations producing "active" branches. It is also important to check the end of every "active" branch in order to "inactivate" it when arriving at its final node. The procedure finishes when all branches are inactive, producing the optimal numbering we were looking for. The flow diagram is shown in Fig. 1.

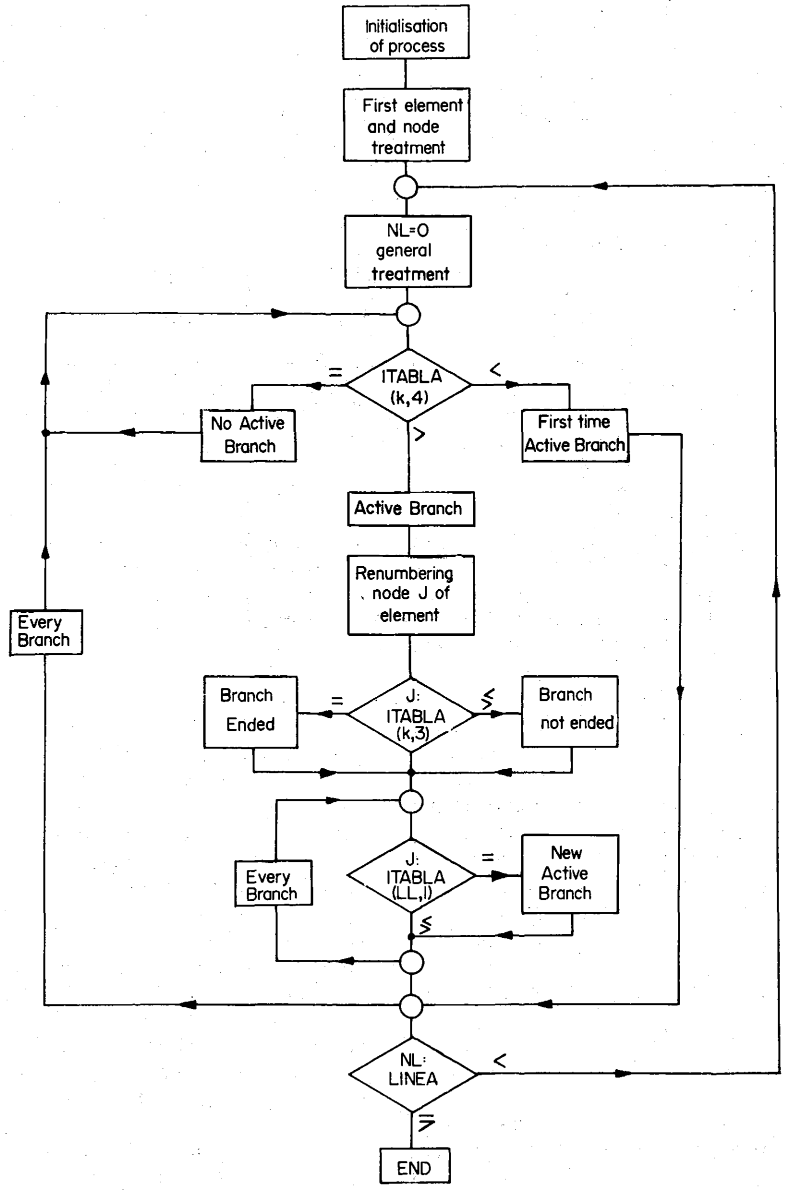

Fig. 1.

3. EXERCISE

As an application of the method, we present a typical example of a line of control pipe whose schematic representation is Fig. 2. The line has 32 nodes and 9 branches, the nodes having been numbered arbitrarily in order to collect all the possibilities appearing in a "real-life" case. The input is that shown in Figs. 3 and 4, with a maximum half bandwith of 156 and a mean bandwidth of 38.9.

The results obtained are the new numbering of members and nodes as shown in Fig. 5 and 6 as well as the specific numbering for every effective degree of freedom in the system. The new maximum half bandwidth is 48 and the mean bandwidth 22.1, which shows the effectiveness of the method in totally arbitrary cases.

4. CONCLUSION

An algorithm for optimal numbering of tree structures has been developed, implementing it with a subroutine which can be used in standard form. It is worth noting the effectiveness of the method, which can be seen in the drastically reduced bandwidth of the example presented in this article (30% of the initial one).

This procedure is currently used as a part of a program (ANPI) designed for dynamic analysis of network pipes and, until now, has been working satisfactorily.

GLOSSARY

IDEN (NNUDOS, 6)	Integer set dimensioned to the node number of the structure; x number of possible degrees of freedom in every node (≤ 6)
NCONE (NBAR, 2)	integer set dimensioned to the element numbers of the structure; x number of nodes per element (2)
ITABLE (NRAMA, 4)	integer set dimensioned to the number of branches of the structure; $x = 4$
LM (NBAR, 13)	integer set dimensioned to the number of elements of the structure; $x = 13$

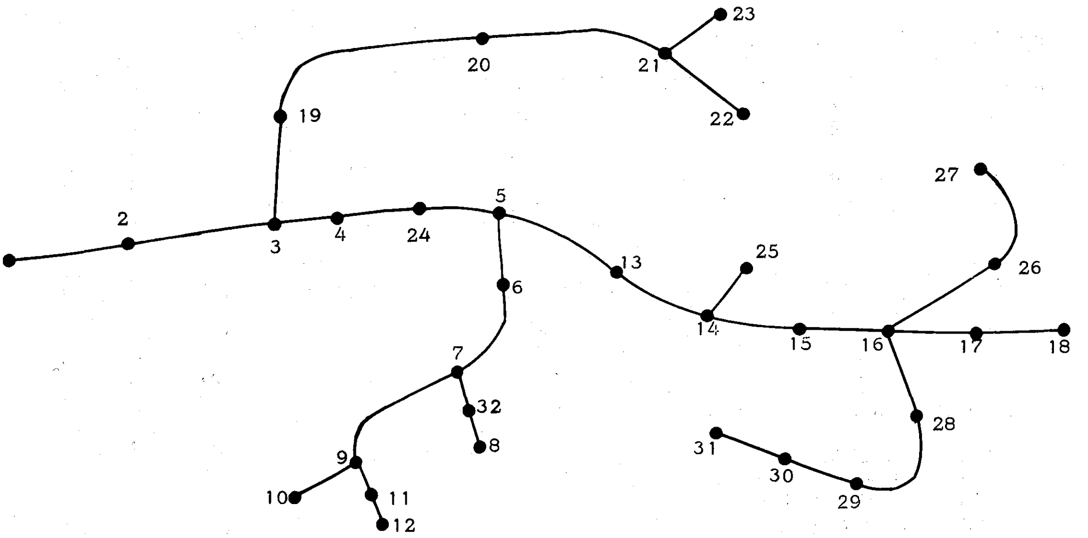

Fig. 2.

```
Q$Q$Q$*AMNAMN( 1).DPNUMER
    1          9    31     3
    2        1  2
    3        2  3
    4        3  4
    5        4 24
    6       24  5
    7        5 13
    8       13 14
    9       14 15
   10       15 16
   11       16 17
   12       17 18
   13       14 25
   14       16 26
   15       26 27
   16        3 19
   17       19 20
   18       20 21
   19       21 22
   20       21 23
   21        5  6
   22        6  7
   23        7  9
   24        9 10
   25        9 11
   26       11 12
   27        7 32
   28       32  8
   29       16 28
   30       28 29
   31       29 30
   32       30 31
   33         23111111
   34          5111000
   35         30100100
   36          1     1    18
   37         14    12    25
   38         16    13    27
   39          3    15    22
   40         21    19    23
   41          5    20    10
   42          9    24    12
   43          7    26     8
   44         16    28    31
```

Fig. 3. INPUT for the MAIN PROGRAM

INPUT DATA

NUM. OF BRANCHES = 9
NUM. OF ELEMENTS = 31
NUM. OF NODES = 32
NUM. OF SUPPORTS = 3

BRANCHES DEFINITION

BRANCH	INIC NODE	INIC ELEM	FINAL NODE
1	1	1	18
2	14	12	25
3	16	13	27
4	3	15	22
5	21	19	23
6	5	20	10
7	9	24	12
8	7	26	8
9	16	28	31

SUPPORTS COND.

NODE	FREEDOMS
5	1 1 1 0 0 0
23	1 1 1 1 1 1
30	1 0 0 1 0 0

MAX. HALF BANDWIDTH= 156

MED. HALF BANDWIDTH= 38.903

TABLE OF CONECTIVITY

ELEM.	I	J
1	1	2
2	2	3
3	3	4
4	4	24
5	24	5
6	5	13
7	13	14
8	14	15
9	15	16
10	16	17
11	17	18
12	14	25
13	16	26
14	26	27
15	3	19
16	19	20
17	20	21
18	21	22
19	21	23
20	5	6
21	6	7
22	7	9
23	9	10
24	9	11
25	11	12
26	7	32
27	32	8
28	16	28
29	28	29
30	29	30
31	30	31

Fig. 4.

OUTPUT

EFECTIVE D.G.F.= 181

ELEMENTS NUMBERS

NEW	OLD	************NODE I************						************NODE J************					
1	1	1	2	3	4	5	6	7	8	9	10	11	12
2	2	7	8	9	10	11	12	13	14	15	16	17	18
3	3	13	14	15	16	17	18	19	20	21	22	23	24
4	15	13	14	15	16	17	18	25	26	27	28	29	30
5	4	19	20	21	22	23	24	31	32	33	34	35	36
6	16	25	26	27	28	29	30	37	38	39	40	41	42
7	5	31	32	33	34	35	36	0	0	0	43	44	45
8	17	37	38	39	40	41	42	46	47	48	49	50	51
9	6	0	0	0	43	44	45	52	53	54	55	56	57
10	18	46	47	48	49	50	51	58	59	60	61	62	63
11	19	46	47	48	49	50	51	0	0	0	0	0	0
12	20	0	0	0	43	44	45	64	65	66	67	68	69
13	7	52	53	54	55	56	57	70	71	72	73	74	75
14	21	64	65	66	67	68	69	76	77	78	79	80	81
15	8	70	71	72	73	74	75	82	83	84	85	86	87
16	12	70	71	72	73	74	75	88	89	90	91	92	93
17	22	76	77	78	79	80	81	94	95	96	97	98	99
18	26	76	77	78	79	80	81	100	101	102	103	104	105
19	9	82	83	84	85	86	87	106	107	108	109	110	111
20	23	94	95	96	97	98	99	112	113	114	115	116	117
21	24	94	95	96	97	98	99	118	119	120	121	122	123
22	27	100	101	102	103	104	105	124	125	126	127	128	129
23	10	106	107	108	109	110	111	130	131	132	133	134	135
24	13	106	107	108	109	110	111	136	137	138	139	140	141
25	25	118	119	120	121	122	123	142	143	144	145	146	147
26	28	106	107	108	109	110	111	148	149	150	151	152	153
27	11	130	131	132	133	134	135	154	155	156	157	158	159
28	14	136	137	138	139	140	141	160	161	162	163	164	165
29	29	148	149	150	151	152	153	166	167	168	169	170	171
30	30	166	167	168	169	170	171	0	172	173	0	174	175
31	31	0	172	173	0	174	175	176	177	178	179	180	181

Fig. 5.

```
NODE NUMBERS
   NEW       OLD          D.G.F. NUMBERS
    1         1        1     2     3     4     5     6
    2         2        7     8     9    10    11    12
    3         3       13    14    15    16    17    18
    4         4       19    20    21    22    23    24
    5        19       25    26    27    28    29    30
    6        24       31    32    33    34    35    36
    7        20       37    38    39    40    41    42
    8         5        0     0     0    43    44    45
    9        21       46    47    48    49    50    51
   10        13       52    53    54    55    56    57
   11        22       58    59    60    61    62    63
   12        23        0     0     0     0     0     0
   13         6       64    65    66    67    68    69
   14        14       70    71    72    73    74    75
   15         7       76    77    78    79    80    81
   16        15       82    83    84    85    86    87
   17        25       88    89    90    91    92    93
   18         9       94    95    96    97    98    99
   19        32      100   101   102   103   104   105
   20        16      106   107   108   109   110   111
   21        10      112   113   114   115   116   117
   22        11      118   119   120   121   122   123
   23         8      124   125   126   127   128   129
   24        17      130   131   132   133   134   135
   25        26      136   137   138   139   140   141
   26        12      142   143   144   145   146   147
   27        28      148   149   150   151   152   153
   28        18      154   155   156   157   158   159
   29        27      160   161   162   163   164   165
   30        29      166   167   168   169   170   171
   31        30        0   172   173     0   174   175
   32        31      176   177   178   179   180   181


        NEW MAX. HALF BANDWIDTH=    48

        NEW MED. HALF BANDWIDTH= 22.129

 CPU TIME FOR RENUMBERING(SEC )=     .0069999695
```

Fig. 6.

N BAR number of structure elements

N NUDOS number of structure nodes

N RAMA number of structure branches

N TGL total number of effective degrees of freedom of the structure

IDEN (I, J) identifies the $J(\leq 6)$ degree of freedom of node I
initially $=0$ d.o.f. J is real
$\neq 0$ d.o.f. J is constricted
finally $=0$ d.o.f. J has been cancelled out and is not considered
$\neq 0$ d.o.f. J is effective and the assigned number in renumbering is IDEN (I, J)

NCONE (I, 1) number initially assigned to starting node of I element

NCONE (I, 2) number initially assigned to final node of I element

I TABLE (I, 1) number initially assigned to starting node of I branch

I TABLE (I, 2) number initially assigned to starting element of I branch

I TABLE (I, 3) number initially assigned to final node of I branch

I TABLE (I, 4) $=0$when branch is not active while computations are being carried out
$\neq 0$ number of element to work with, when the branch is active

LM (I, 1) number finally assigned to I element

LM (I, 2) $\div$ LM (I, 7) finally assigned numbers to the effective degree of freedom of initial node of I element

LM (I, 8) $\div$ LM (I, 13) finally assigned numbers to the effective degree of freedom of final node of I element

Acknowledgements—This work is a part of the ANPI program (static and dynamic analysis of pipes) prepared for the firm SAINCO. Our thanks are especially directed to J. Quiroga and F. Delgado, members of the technical staff of the firm.

REFERENCES

1. ANPI, *Theoretical and User's Manual.* SAINCO, Madrid, (1979).
2. A. Jennings, *Matrix Computation for Engineers and Scientists.* Wiley, New York (1977).
3. J. K. Reid, Mesh generation. *Proc. F. E. Symp.*, London. March (1975).
4. NUPIPE, *User's Manual.* Nuclear Services Corporation. California.
5. STRUDL. *User's Manual*, IBM.
6. L. Durscher and A. Gasper, A versatile two-dimensional mesh generator with automatic bandwidth reduction. *Comp-Struc.* **10**, 561–575 (1975).
7. G. C. Everstine, A comparison of three resequencing algorithms for the reduction of matrix profile and wavefront. *Int. J. Num. Mat.* **14**, 837–853 (1979).

APPENDIX

```
@PRT,S AMNAMN.NUMER
FURPUR 27R2  RL72R1 07/29/80 13:29:26
Q$Q$Q$*AMNAMN(1).NUMER
            SUBROUTINE NUMER(IDEN,NCONE,ITABLA,NRAMA,LM,NTGL)
      C
      C
      C      E.T.S. INGENIEROS INDUSTRIALES
      C      MADRID (SPAIN)
      C
      C     SUBROUTINE FOR RENUMBERING OF EFECTIVE DEGREES OF FREEDOM
      C     IN TREE STRUCTURES
      C
            DIMENSION IDEN(51,6),NCONE(50,2),ITABLA(10,4),LM(50,13)
      C
      C
      C     INITIALISATION OF PROCESS
      C     TREATMENT OF THE FIRST ELEMENT AND BRANCH
      C
            NTGL=0
            IBAR=0
            ITABLA(1,4)=1
      C
      C     NUMBERING THE EFECTIVE D.G.F. OF NODE 1
      C
            DO 10 I=1,6
            IF(IDEN(1,I))11,12,11
      C
      C     EFECTIVE D.G.F.
      C
         12 NTGL=NTGL+1
            IDEN(1,I)=NTGL
            GO TO 10
      C
      C     RESTRICTED D.G.F.
      C
         11 IDEN(1,I)=0
         10 CONTINUE
      C
      C     GENERAL PROCESS
      C
        300 NL=0
            DO 200 K=1,NRAMA
            IF(ITABLA(K,4))21,22,23
      C
      C     BRANCH FIRST TIME ACTIVE
      C
         21 ITABLA(K,4)=-ITABLA(K,4)
            GO TO 200
      C
      C     UNACTIVE BRANCH
      C
         22 NL=NL+1
            GO TO 200
      C
      C     ACTIVE BRANCH
      C
         23 M=ITABLA(K,4)
            I=NCONE(M,1)
            J=NCONE(M,2)
            IBAR=IBAR+1
      C
      C     NUMBERING NODE J OF THE ELEMENT AND
      C     STORAGE ITS EFECTIVE CONECTIVITIES
      C
            DO 25 LL=1,6
            IF(IDEN(J,LL))26,27,26
      C
      C     EFECTIVE D.G.F.
      C
         27 NTGL=NTGL+1
            IDEN(J,LL)=NTGL
            GO TO 28
      C
      C     RESTRICTED D.G.F.
      C
         26 IDEN(J,LL)=0
      C
      C     EFECTIVE CONECTIVITY OF ELEMENT M
      C
```

```
   28 LM(IBAR,LL+1)=IDEN(I,LL)
   25 LM(IBAR,LL+7)=IDEN(J,LL)
      LM(IBAR,1)=M
      IF(J-ITABLA(K,3))31,30,31
C
C     ENDED BRANCH. IT'S UNACTIVED
C
   30 ITABLA(K,4)=0
      GO TO 32
C
C     NOT ENDED BRANCH
C
   31 ITABLA(K,4)=ITABLA(K,4)+1
   32 CONTINUE
C
C     SEARCHING NEW ACTIVE BRANCHES
C
      DO 40 LL=1,NRAMA
      IF(J.NE.ITABLA(LL,1)) GO TO 40
C
C     NEW ACTIVE BRANCH
C
      ITABLA(LL,4)=-ITABLA(LL,2)
   40 CONTINUE
  200 CONTINUE
      IF(NL.LT.NRAMA) GO TO 300
C
C     END OF PROCESS
C
      RETURN
      END
C
C     E.T.S. INGENIEROS INDUSTRIALES
C     MADRID (SPAIN)
C
C     MAIN PROGRAM FOR RENUMBERING OF TREE STRUCTURES
C
      DIMENSION NCONE(50,2),IDEN(51,6),ITABLA(10,4),LM(50,13),LB(6
         COMMON/CPUT/TCPU
C
C     READ INPUT DATA
C
      LEC=5
      IMP=6
      READ(LEC,1)NRAMA,NBAR,NAPO
    1 FORMAT(4I5)
      NNUDOS=NBAR+1
      READ(LEC,2)((NCONE(I,J),J=1,2),I=1,NBAR)
    2 FORMAT(2I3)
        DO 100 K=1,NAPO
        READ(LEC,3)I,LB
    3   FORMAT(I5,6I1)
          DO 150 J=1,6
  150     IDEN(I,J)=LB(J)
  100   CONTINUE
      READ(LEC,1)((ITABLA(I,J),J=1,4),I=1,NRAMA)
C
C     WRITE INPUT DATA
C
      WRITE(IMP,11)NRAMA,NBAR,NNUDOS,NAPO
   11 FORMAT(1H1//5X,'INPUT DATA'//
     *5X,'NUM. OF BRANCHES  =',I5/
     *5X,'NUM. OF ELEMENTS  =',I5/
     *5X,'NUM. OF NODES     =',I5/
     *5X,'NUM. OF SUPPORTS  =',I5)
      WRITE(IMP,12)
   12 FORMAT(//5X,'TABLE OF CONECTIVITY '//5X,'ELEM.     I     J')
      WRITE(IMP,13)(I,(NCONE(I,J),J=1,2),I=1,NBAR)
   13 FORMAT(5X,3I5)
      WRITE(IMP,14)
   14 FORMAT(//5X,'SUPPORTS COND.'//5X,' NODE     FREEDOMS')
      DO 110 I=1,NNUDOS
      K=0
      DO 115 J=1,6
      IF(IDEN(I,J).NE.0) K=1
  115 CONTINUE
      IF(K.EQ.1)
     *WRITE(IMP,15)I,(IDEN(I,J),J=1,6)
  110 CONTINUE
```

```
      SUM=0.
      NBAN=0
      DO 120 K=1,NBAR
      NDIF=ABS(NCONE(K,1)-NCONE(K,2))
      IF(NDIF.GT.NBAN) NBAN=NDIF
      SUM=SUM+(NDIF+1)*6
  120 CONTINUE
      SUM=SUM/NBAR
      NBAN=(NBAN+1)*6
      WRITE(IMP,18)NBAN,SUM
   18 FORMAT(//10X,'MAX. HALF BANDWIDTH=',I5//
     *10X,'MED. HALF BANDWIDTH=',F7.3)
   15 FORMAT(5X,I5,3X,6I2)
      WRITE(IMP,16)
   16 FORMAT(//5X,'BRANCHES DEFINITION     '//5X,'BRANCH
     *' INIC NODE   INIC ELEM     FINAL NODE')
      WRITE(IMP,17)(I,(ITABLA(I,J),J=1,3),I=1,NRAMA)
   17 FORMAT(5X,I5,3I12)
C
C       CALL RENUMBERING SUBROUTINE
C
         CALL CPU
         TTTT=TCPU
      CALL NUMER(IDEN,NCONE,ITABLA,NRAMA,LM,NTGL)
         CALL CPU
         TTTT=TCPU-TTTT
         WRITE(IMP,1111)TTTT
1111     FORMAT(////5X,'CPU TIME FOR RENUMBERING(SEC )=',F15.10///)
C
C WRITE OUTPUT
C
      WRITE(IMP,20)NTGL
   20 FORMAT(1H1///10X,'OUTPUT'//5X,'EFECTIVE D.G.F.=',I5)
      WRITE(IMP,21)
   21 FORMAT(//5X,'ELEMENTS NUMBERS'/5X,'       NEW     OLD',
     *5X,12('*'),'NODE I',12('*'),5X,12('*'),'NODE J',12('*'))
      WRITE(IMP,22)(I,(LM(I,J),J=1,13),I=1,NBAR)
   22 FORMAT(5X,2I8,5X,6I5,5X,6I5)
      WRITE(IMP,23)
   23 FORMAT(//5X,'NODE NUMBERS'/5X,'       NEW      OLD',
     *8X,'D.G.F. NUMBERS')
      I=1
      WRITE(IMP,24)I,I,(IDEN(I,J),J=1,6)
        DO 200 I=2,NNUDOS
        K=LM(I-1,1)
        J=NCONE(K,2)
  200   WRITE(IMP,24)I,J,(IDEN(J,K),K=1,6)
   24   FORMAT(5X,2I8,5X,6I5)
      SUM=0
      NBAN=0
      DO 130 K=1,NBAR
      I=0
      J=0
      DO 135 L=2,13
      IF(LM(K,L))135,135,136
  136 I=LM(K,L)
      GO TO 137
  135 CONTINUE
  137 DO 140 L=13,2,-1
      IF(LM(K,L))140,140,141
  141 J=LM(K,L)
      GO TO 142
  140 CONTINUE
  142 NDIF=J-I+1
      SUM=SUM+NDIF
      IF(NDIF.GT.NBAN) NBAN=NDIF
  130 CONTINUE
      SUM=SUM/NBAR
      WRITE(IMP,19)NBAN,SUM
   19 FORMAT(//10X,'NEW MAX. HALF BANDWIDTH=',I5//
     *10X,'NEW MED. HALF BANDWIDTH=',F7.3)
      STOP
      END
```